TLN - DiCaro

Mario Bifulco

a.a. 2022-2023

Indice

1 Semantica computazionale

Semantica lessicale Studia come sono fatte le parole (il lessico)
Semantica formale Riguarda la definizione dei linguiggi logico-formali
Semantica statistica Semantica basata sulla statistica

Semantica linguistico-distribuzionale Insieme di modelli che descrivono la distribuzione semantica delle parole nel linguaggio sia in modo statistico che linguistico

Il TextMining è un approccio puramente statistico di analisi semantica, ad esso fa riferimento il sottoramo della semantica documentale (ovvero legata a collezioni di documenti)

I primi tentativi di NLP riguardavano la risposta a domande poste da un utente riferite ad un determinato testo. Ci sono diverse possibili domande che si possono porre al sistema, tali domande possono essere catalogate e in base alla categoria è necessario reperire diverse informazioni semantiche. Il task di question answering è stato a lungo considerato troppo complesso e quindi intrattabile, per questo motivo i ricercatori hanno diviso il problema in blocchi fondanti e lavorato su questi (NER tagging, relazioni tra sensi, WSD, suggeritori, LM)

2 Significato del significato

Lessico Dizionario di elementi a disposizione per costruire la frase

Sintassi Relazioni tra gli elementi attraverso le quasi si costruisce una struttura e quindi una frase

Semantica Interpretazione della struttura e assegnamento di un significato

Pragmatica Uso contestuale/situazionale della lingua

Ambiguità Proprietà del linguaggio, permette una comunicazione efficiente

Polisemia Una parola può esprimere più significati

Omonimia Parole con la stessa forma ma significati diversi

Comunicazine Condivisione di significati che risiedono nella mente

Convenzione Veicolo del contenuto semantico tramite simboli (es. suoni)

Granularità Livello di dettaglio che si vuole descrivere

Soggettività Essendo il linguaggio un'approssimazione delle immagini della nostra mente si possono commettere errori durante l'esposizione

Similarità Meccanismo che permette l'inferenza di significato da altri termini conosciuti

Esperienza personale Insieme di eventi che forma il nostro modo di comunicare

Senso comune Convenzioni a livello collettivo

Cultura Convenzione storica/senso comune storico

Tre principali teorie per determinare il significato ("word meaning"):

1. Basate su primitive (Kats, Wilks, Lakoff): per rappresentare il significato di una parola si frammentano i contenuti semantici in atomi

2. Basate su relazioni (Quinlan, Fodor): il significato è frutto delle relazioni con le altre parole (una parola di per se non ha significato)

3. Basate su composizioni (Pustejovski): il significato si basa non solo sulle parole vicine, ma anche sulla loro composizione

Ognuna di queste teoria è "attaccabile"

1. Non tutto è descrivibile per somma di primitive (problema olistico)

2. Noi associamo istintivamente un significato ad una parola, anche non avendo contesto

3. Come si trattano lingue in cui la composizione non conta? (Warlpiri)

Triangolo semiotico Modello del significato, ogni concetto mentale è rappresentabile con un triangolo concetto, referente e rappresentazione. Il concetto è ciò a cui stiamo pensando e che vorremmo comunicare (rappresentazione extralinguistica). La rappresentazione include nel concetto la convenzione per comunicarlo. Il referente è l'individuo con cui vogliamo comunicare. I sistemi automatici partono dalla rappresentazione (es. il Web) per creare una concettualizzazione per poi, eventualmente, focalizzarsi sui referenti

Il multilinguismo è sia un problema che una possibilità, problema perché va gestito, possibilità perché permette uno studio semantico più ricco, in quanto ogni lingua può catturare diverse sfumature di significato

I concetti possono avere diversa granularità con cui vengono espressi, singole parole, chunk, frasi, discorsi, documenti, collezioni di documenti

Il task di WSD ha una serie di problemi, come:

- Specificità: le risorse come WordNet risultano troppo polverizzate per molti contesti applicativi

- Copertura: le risorse potrebbero avere intere aree del linguaggio non coperte

- Soggettività: le decisioni di annotazione non sono prese in modo autometico e standard

Differenze tra WSD e WSI

- WSD ha bisogno di sense inventory per disambiguare, WSI cerca invece di ricavare il senso partendo dalla sola collezione di testi
- WSI si basa sull'uso della parola e non sulla risorsa specifica
- WSD è molto basata sulla grammatica della lingua
- WSD è molto semplice da valutare, ma bisogna fare attenzione al problema della polisemia
- La WSI si può valutare tramite pseudoword, si scelgono due parole, si sostituiscono con una nuova parola (inesistente nel dizionario d'origine), WSI deve riuscire a creare due cluster di significato

Per cercare significati sono disponibili dizionari elettronici come WordNet e BabelNet, risorse linguistico cognitive come Property norms (studi cognitivi sulle dinamiche del linguaggio), common-sense knowledge (come ConceptNet), Visual Attributes (risorse per descrivere la fisionomia degli oggetti), Corpus manager. Queste risorse raccolgono tra le varie cose le definizioni, frasi complesse da scrivere in quanto non si ha un modo uniforme e codificato per descrivere un concetto, identificare le proprietà principali, capire le relazioni semantiche o per valutare la qualità di una definizione

3 Costruzione del significato

Generative lexcon di Pustejovski Teoria semantico linguistica basata su:

- Argument structure: legame tra sintassi e semantica del concetto
- Event structure: eventi che hanno il concetto che si vuole esprimere come stato
- Qualia structure: struttura del concetto/caratteristiche
- Inheritance structure: posizionamento in una tassonomia di sensi

Questo sistema permette un ragionamento preciso e completo su tutte le clasi semantiche di ragionamento, il problema è che risulta complessa da implementare e da computare.

Ruoli dei qualia:

- Ruolo costruttivo: parte composizionale, rappresenta le caratteristiche materiali (peso, dimensione, parti componenti)
- Ruolo formale: caratteristiche definenti (cosa lo contraddistingue nel suo dominio di appartenenza?)
- Ruolo telico: obiettivo o funzione del concetto, ruolo comportamentale
- Ruolo agentive: serie di entità o eventi naturali che rappresentano l'origine del concetto

Ruoli qualia di Hanks Teorizza la teoria delle Valenze, il verbo è la radice del significato e i sostantivi a contorno si incastrano per comporre un significato più ampio. La valenza è la cardinalità degli elementi vicini. Hanks introduce i concetti di collocazione e tipo semantico. La collocazione è la combinazione dei possibili filler intorno al verbo. Il tipo semantico è la macro categoria che ci si aspetta di trovare, la definizione dei tipi semantici non è semplice. I tipi semantici possono essere più o meno specifici in base al verbo, ma capire quando è necessario un certo livello di dettaglio è un problema aperto

Affordance linguistiche Affordance è un termine introdotto da Gibson, un oggetto è percepibile anche se non si è mai visto prima. In linguistica si applica a quanto facilmente la parola è autoesplicativa

Altro concetto utile se si parla di sforzo cognitivo o computazionale per generare del testo sono i pattern, che risultano standard e facili da riempire, al contrario un discorso che non segue pattern standard può essere comunque corretto, ma meno fluido alla comprensione

I dati per questo tipo di annotazioni (che poi ricadono nei cluster e nei tipi semantici di Hanks) si possono ricavare da questionari, proprietà latenti della similarità, risorse lingusitiche come corpora, annotazioni manuali, IE, ecc.. Un vantaggio dei cluster basati sulla descrizione delle proprietà (e quindi non decisi a priori) è che bastano pochi esempi per decidere le feature caratteristiche (ulteriori esempi non aggiungerenno informazioni altamente discriminanti)

4 Text Mining

Si tratta del ramo di ricerca che vuole applicare le tecniche di data mining al testo. Per questo tipo di analisi si passa da un approccio top-down

(tipico della linguistica) ad un approccio bottom-up (proprio della statistica). In questa analisi le unità fondamentali non sono le parole ma i token, ovvero parti di parole particolarmente frequenti. Rappresentando i testi come una sequenza (e relativa frequenza) di token si può passare alla formalizzazione vettoriale. Tale vettore è formato indicizzando il dizionario dei token possibili, si procede poi a contare i token presenti nel testo. Da una collezione di documenti si può ottenere la rappresentazione matriciale, tale matrice descriverà in modo univoco quella collezione e sarà caratterizzata da valori sparsi. Il vector space model, ovvero la matrice sparsa, ha permesso di effettuare ricerche in modo estremamente più rapido tramite l'uso della cosine-similarity

Oltre al metodo classico per la rappresentazione vettoriale si possono usare i metodi statistici, in particolare il metodo basato su frequenze e quello su co-occorrenze. Il metodo basato sulle frequenze si concentra su quanto il termine sia frequente, e quindi significativo, all'interno di un testo e quanto sia raro a livello di collezione (TF-IDF), da notare che per svolgere confronti ragionevoli la frequenza dovrà essere normalizzata per la lunghezza del testo, l'idf invece viene compresso tramite il logaritmo. Il metodo delle co-occorrenze invece cattura quanto due parole sono simili, ovvero è probabile compaiano negli stessi contesti, questo tipo di rappresentazione permette un cosine-similarity più robusta, in quanto sinonimi morfologicamente diversi ma concettualmente uguali compariranno nello stesso contesto.

Alcune applicazioni si basano sulla mera rappresentazione dei dati, ad esempio le Tag-Cloud, in base alla frequenza delle parole sono stampati i termini più frequenti con dimensione crescente, questo metodo di visualizzazione permette di capire velocemente i termini i più frequenti e in modo approssimativo il "topic" del testo analizzato, la Tag-Cloud può anche essere usata con le co-occorrenze, in questo caso termini vicini corrispondono a termini che compaiono negli stessi contesti. Altra possibile applicazione sono le Tag-Flakes, che includono le co-occorrenze, l'idea è estrarre in generale una gerarchia di termini. Si calcola il rank della parola in base alla frequenza e poi si cerca di estrarre il topic in base alla co-occorrenza. Potrebbe anche essere utile effettuare il clustering dei documenti per poi visualizzarli in sotto-gruppi, esistono vari metodi di clustering e sono da valutare in base alla specifica situazione. Oltre al clustering è possibile classificare i testi e determinare le etichette relative. Altro task che utilizza Text Mining è quello della segmentazione di un testo, si vogliono separare i diversi contenuti da un testo sequenziale. Un esempio di segmentation è l'algoritmo di TextTiling, che divide i testi in base alla coesione intra-segmento e alla non coesione inter-segmento, in un loop vengono divisi i testi, calcolate le coesioni, ricercati eventuali punti migliorativi della situazione attuale e riadattati i tagli.

Il riassunto automatico si pone come obiettivo quello di accorciare un testo di partenza mantenendone il significato, i riassunti possono essere estrattivi (prendo le frasi più significative, TextRank) o astrattivi (genero un testo del tutto nuovo). La valutazione dei riassunti può essere fatta tramite la metrica ROUGE (sistema che mappa gli n-grammi tra un testo di riferimento e quello ottenuto). Infine le tecniche di text mining possono essere usate per Information Retrieval, ad oggi per fare information retrieval non si usano più solo i testi ma informazioni più sofisticate.

5 Semantica distribuzionale

Nella semantica distribuzionale a differenza delle tecniche di TextMining ci si concentra maggiormente sulla linguistica.

- Harris: parole che appaiono in contesti simili hanno significato simile
- Firth: una parola è caratterizzata dalla sua compagnia
- Furnas: l'uso congiunto di più parole permette di esprimere meglio i concetti
- Deerwester: esistono i concetti latenti, struttura di base parzialmente perse quando si parla in un discorso
- Blei: l'argomento influenza le parole scelte per usare per descriverlo
- Turney: coppie di parole con modelli simili hanno relazioni semantiche simili

Anche in questo caso vengono usate matrici per le rappresentazioni, questo perché le matrici sono approssimazioni, come il linguaggio stesso. Le matrici sono a metà strada tra una rappresentazione simbolica e una rappresentazione associazionistica/connessionistica

Per lavorare al meglio con le rappresentazioni matriciali (indipendentemente dal singolo caso) è utile applicare alcune operazioni di pre-processing. Tra queste troviamo la normalizzazione (e la controparte denormalizzazione) un processo che restringe la variabilità del linguaggio, ad esempio tramite lemmatizzazione.

Alcune rappresentazioni matriciali utilizzate sono:

1. Term-Document matrix: vede i documenti come serie di termini (righe = documenti, colonne = termini)

2. Term-Context matrix: riga = contesto, colonna = termine, generalizzazione del term-document

3. Pair-Pattern matrix: riga = coppie di parole, colonne = pattern (X è causa di Y), questo permette di calcolare la relationa-similarity

La similarità è fondamentale per l'apprendimento (lo dice Quine a partire dagli anni 60) e si possono distinguere diversi tipi:

1. Semantic Similarity: concetti che hanno quasi lo stesso significato, sinonimi

2. Semantic Relatedness: concetti che dividono delle proprietà, non viene usata perché restituisce solo il fatto che due concetti sono in relazione, senza specificare quale

3. Attributional Similarity: similarità tra concetti che sono attributi

4. Taxonomical Similarity: similairtà tra ipernonimi

5. Relational Similarity: similarità tra coppie o tuple di concetti

6. Semantic Association: concetti che co-occorrino di frequente

Il problema della rappresentazione matriciale è che viene perso il concetto di ordine delle parole. La rappresentazione tramite matrici permette di raggiungere al massimo 80% di accuratezza, questo risultato è buono per IR. Per attenuare il problema si ricorre a matrici pair-pattern, più sensibili all'ordinamento, o a vettori ausiliari che catturano l'ordinamento. Altro problema è che la rappresentazione matriciale non è composizionale, per cercare di colmare questo gap si cercano vettori semplici da combinare per avere vettori più complessi.

6 Semantica documentale

Ricerca a livello di collezione di documenti. Il topic modelling è un modello statistico che individua gli argomenti della collezione. Rientra nei modelli non supervisionati. L'estrazione dei topic non è sempre così ovvia, in linea di massima sono estratti in base alle frequenze delle parole nei contesti. La tecnica storica per il topic modelling è LSA (latent semantic analysis), che applica una fattorizzazione matriciale SVD (singular value decomposition). Le tre matrici prodotte tramite SVD corrispondono a: la nuova rappresentazione delle feature, diagonale dei singular values, feature latenti trasposte.

Lo svantaggio è che nel passaggio a vettori densi compaiono anche valori negativi, che non hanno una lettura intuitiva. LSA in generale presenta alcuni problemi, il modello ottenuto non generalizza su documenti non visti, non si possono interpretare i valori negativi (ottenuti per via di SVD). Un evoluzione di LSA è la sua versione probabilistica che si generalizza con LDA (latent dirichlet allocation). LDA usa un approccio statistico bayesiano.

6.1 Text Visualization

Gli approcci esposti sono tutti n-dimensionali, quindi la loro rappresentazione non è immediata. Alcuni metodi di rappresentazione sono:

1. Parallel Coordinates: ogni dimensione è una retta parallela, i vettori sono descritti come spezzate che collegano vari punti di queste linee verticali (in base al loro valore per la coordinata i)
2. RadViz: le dimensioni sono punti della ciconferenza, all'interno sono rappresentati i documenti in base alla vicinanza con le classi
3. HeatMap: i dati sono espressi come matrice e il colore indica il valore numerico
4. Correlation Circle: nella circonferenza troviamo gli elementi e verso l'interno i collegamenti/co-occorrenze

7 Text2Everything

Le macchine possono apprendere tramite:

1. Apprendimento supervisionato: il dataset ha l'etichetta target
2. Apprendimento semi-supervisionato: il modello impara da dati etichettati e impara a generalizzare da dati non etichettati
3. Apprendimento debole supervisionato: le etichette contengono rumore o sono imprecise
4. Apprendimento auto-supervisionato: il modello genera la propria supervisione partendo dall'input
5. Apprendimento per rinforzo: il modello impara tramite feedback

Queste tecniche possono anche essere combinate per i processi di pre-training e fine-tuning. Durante il pre-training i modelli sono addestrati e i pesi vengono salvati. Successivamente si usano questi pesi per l'inizializzazione della rete a cui si vuole fare fine-tuning. Il vantaggio di questa procedura è che permette di lavorare con dataset diversi, mantenendo i dati realmente di interesse (che spesso sono pochi) solo per l'ultima fare, inoltre addestrare un modello pre-trained è meno onoreso rispetto all'inizializzazione casuale dei pesi. Alcune tra le tecniche più usate per il pre-training sono:

1. Masked Language Modeling (MLM): alcune parole sono cancellate e il modello deve riempire i buchi (apprende sintassi e struttura del linguaggio)

2. Denoising AutoEncoder (DAE): introduce rumore nei dati, il modello deve ricostruire l'input originale (cattura le caratteristiche salienti dei dati)

3. Replaced Token Detection(RTD): task discriminativo, il modello deve riconoscere se un token è stato generato da un modello di linguaggio (apprende la coerenza del testo)

4. Next Sentence Prediction (NSP): date due frasi il modello deve determinare l'ordine corretto (apprende la correlazione tra due frasi)

I modelli aperti sono più vantaggiosi, questo perché garantiscono personalizzazione (il singolo sviluppatore può adattarlo alle proprie esigenze), privacy (il singolo sviluppatore può mantenere dati sensibili in locale, senza esporli ad eventuali terzi), riduce i costi, rende indipendenti dai fornitori inoltre incentivano la ricerca

Uno dei metodi largamente utilizzati per la fase di fine-tuning è il Reinforcement Learning from Human Feedback (RLHF), l'agente apprende tramite rinforzo e un algoritmo di ottimizzazione (come PPO, proximal policy optimization), annotatori umani valutano i vari output, classificandoli come desiderabili o meno, la valutazione dell'output viene fatta tramite un sistema ELO. In breve i modelli RLHF prendono il LLM pre-trained, generano un reward model e usano RL per il fine tuning. Il RL è caratterizzato da una policy (coppia prompt-testo di risposta), spazio d'azione (token del LM), spazio d'osservazione (distribuzione dei possibili output) e funzione di ricompensa (combinazione tra preference model e vincoli sul cambiamento della policy). Il calcolo della preferenza dell'output vieen fatta tramite Kullback-Leibler (KL) divergence. I modelli basati su RLHF richiedono molte persone

per avere risultati notevoli, inoltre l'etica del modello viene messa in mano ai giudici, che possono avere sentisibilità diverse rispetto ai potenziali utilizzatori, inoltre hanno un costo operativo oltre che computazionale

Una possibile soluzione è la Constitutional AI, in cui si prende il modello prodotto tramite RLHF, questo modello risponde a molte domande (potenzialmente dannose) generando una bozza, il sistema mostra al modella la bozza e un prompt fittizio (es. riscrivi in modo etico). Questo processo viene ripetuto per ottenere un dataset bozze-riscritture etiche, il modello di partenza viene addestrato sul nuovo dataset Quello che si vuole ottenere sono modelli che arrivano alla frontiera di Pareto (ovvero non possono migliorare l'efficacia senza compromettere l'innocuità e vieceversa)

7.1 Uso e fine-tuning LLM

Durante l'uso di un LLM possiamo usare dei parametri. Diversi dagli iperparametri, questi ultimi sono usati durante il training per gestire il comportamento stocarstico. I parametri invece vengono usati per guidare l'output. Un esempio di parametro è proprio la temperatura, che gestisce la creatività del modello, Top p (nucleus sentence), genero le parole e prendo quelle nel percentile Top p migliore, Top k, simile a Top k ma a grana grossa (gestisce il numero di token), Beams, genera n frasi e sceglie la migliore

LORA (low rank adaptation), alternativa al fine tuning. Al posto che fare fine-tuning si inietta alla fine dei layer (full-rank, ovvero linearmente indipendenti) delle matrici "trainable". Nel fine-tuning classico riaddestro la rete sul dataset, con LORA inietto delle matrici singolari (non full-rank) per semplificare la moltiplicazione matriciale. Queste matrici low-rank hanno precisione minore ma allegeriscono il fine-tuning. Si diminuisce di molto la dimensione della rete con prestazioni paragonabili. Soprattutto sui task generativi non siamo interessati a risultati super precisi

Oltre alla riduzione tramite LORA spesso si usa quantization per ridurre il costo computazionale e d'inferenza. Al posto di usare il classico double uso un float più piccolo se non anche un int8. QLoRa usa quantization a 4bit. Bisogna bilanciare la perdita di prestazioni con il vantaggio dimensionale. La quantization non è sempre disponibili, non tutte le GPU ad es. possono ridurre a int8, quindi la quantization migliore si ottiene a float16. Per le CPU invece va bene int8

Txt2Audio NanoGPT, architettura molto piccola per gestire i testi.
Quando si esegue tokenization si associa anche il rumore gutturale e si può riprodurre audio. Il modello non capisce il legame semantica-pronuncia. L'uso di NanoGPT rende problematica la creazione di audio lunghi, in questo

caso è meglio usare il modello LongFormer. Il passaggio da una lingua all'altra è abbastanza semplice proprio perché non c'è reale comprensione della lingua. I modelli possono anche essere usati per fare semplice compressione, i codec neurali come EnCoded di Meta riesce ad emulare le frequenze gradite all'orecchio umano togliendo solo il "superfluo".

Txt2Img Fino ad un paio di anni fa si usavano GAN, buone ma non abbastanza realistiche. I modelli oggi sono basati su DiffusionProcess. I dataset sono estratti dal web, la descrizione va estratta da quella della pagina web, quindi hanno un sacco di rimore, vanno ripuliti per cercare di avere risultati accettabili. Il processo di diffusione sfrutta un'approssimazione della catena di Markov, nella forward pass si distrugge la struttura dei dati tramite rumore gaussiano (i dati sono assorbiti nello spazio latente), nella backward pass si fa denoising dell'immagine. StableDiffusione ha una serie di Transformer in catena, il prompt viene tokenizzato, un modello trasforma in embeddings (ed. CLIPText), il risultato si fa passare nel processo di creazione dell'immagine (UNet + Scheduler), lo stato dell'arte crea immagini ottime in meno di 10 passaggi (generati tramite Scheduler), infine l'image decoder (AutoEncoderDecoder) trasforma il tensore in immagine

Per permettere ai modelli di girare anche su computer normali si effettua attention-slicing, l'attenzione viene segmentata per processare frammenti gestibili dalla ram della GPU

Un parametro classico per i modelli a diffusione è CFG, classifier-free guidance, funziona in modo simile alla temperatura per il testo (tipicamente messo a 7.5). Si possono anche usare prompt negativi per rimuovere determinati embedding

I modelli addestrati su coppie immagine-testo possono essere usati anche per applicazioni zero-shot (es. domande sulle immagini), effettuano "ricerche" sulla base documentale latente, si comportano come se capissero semanticamente

8 Basicness

Gli studi sulla Basicness del linguaggio non sono molti in ambito NLP, tra le prime ricerche troviamo quella di Roger Brown (1958). In questo studio si sofferma su come dovremmo chiamare i concetti (come scegliamo i termini). Già nel 1930 Ogden produce un "vocabolario di base", Brown ha poi sottolineato l'importanza di stabilire quando un termine può riternersi di base o meno. Ricercatori successivi hanno introdotto una gerarchia di termini (base-middle-advanced). Oggi il basic-level viene spesso associato al

concetto di "sopravvivenza sociale", ovvero i primi termini che chi impara la seconda lingua apprende per essere in grado di comunicare. Nonostante la creazione di dizionari di base spesso i termini presenti possono essere usati per esprimere concetti avanzati (es. cane che nelle armi fa riferimento a un pezzo specifico della pistola), inoltre la lingua è in buona parte soggettiva, quindi non sempre è immediato concordare sulla definizione di basicness. Da notare anche non è mai stata proposta una definizione per i termini avanzati, se non come inverso dei termini base

9 Ontology learning

Consiste in un reverse engeneering di quale informazioni viene trasposta tramite scrittura, operazione che va dallo scritto ai modelli mentali. Si distinguino 3 sotto-task principali:

1. Ontology population: collezione di informazioni machine readable, ho già l'ontologia e devo incasellare della nuova informazione testuale nell'ontologia

2. Ontology based annotation: dal documento voglio aggiungere link a pezzi dell'ontologia

3. Ontology enrichment: voglio arricchire un'ontologia partendo da una base documentale (aggiungere, modificare o cancellare nodi e relazioni)

Dal document repository si può scendere a diverse profondità: Estrazioni di termini, Glossari, Thesauri, Tassonomie, Ontologie, Regole logico/inferenziali. Ontology learning può essere fatto con diversi metodi derivanti da NLP, formal concept analysis (FCA) e machine learning. FCA formalizza problemi con approcci deterministici all'estrazione dell'informazione. Il concetto fondamentale è il **formal context**, formato da oggetti, attributi e incidenza (relazione oggetto-attributo). Il Formal Context è una matrice di incidenza. Su questa matrice sono definiti due operatori, uno che vuole estrarre le feature dato il conetto, il secondo prende in input le feature e restituisce la lista dei concetti che soddisfano quelle proprietà. Dalla tabella si può estrarre un lattice (gli estremi sono l'insieme delle feature e l'insieme vuoto di feature), il primo livello prende le feature singolarmente, associando i relativi concetti (matching totale), per i livello successivi si creano combinazioni tra feature, proseguendo così esprimo tutti i concetti e ho l'estensione massima nei livelli centrali del lattice. Il lattice ottenuto è un modo formale per definire un rapporto individui-feature, crea una gerarchia concettuale.

Quindi si può costruire un'ontologia dal Formal Context in modo automatico, ad esempio partendo dal testo e relazioni sintattiche creo l'ontologia che descrive la collezione di testi. In passato è stata molto utilizzata per approcci multilinguistici

10 Open information extraction

Nasce dalla necessità di estrarre informazioni da grandi corpora in modo rapido. Vogliamo conoscenza semi strutturata in forma di triplette (arg1, VP, arg2). La base di conoscenza così costruita era usata per il Q&A

OIE è difficile da valutare, questo crea problemi se si vuole adoperare in larga scala. La difficoltà nella valutazione è anche dovuta ad approcci non standard (es. alcuni lavoravano su n-uple). Non c'era ground truth per comparare estrazioni. A volta viene chiamata Shallow Ontology Learning

11 Prompting con LLM

Un tempo si faceva feature engeneering per gli algoritmi di machine learning. I modelli del linguaggio con approcci deep sono diventati abbastanza abili per svolgere task generici. Quindi c'è stato uno shift verso il prompting (come chiedere le cose per ottenere buoni risultati). Per un buon prompt servono: istruzioni chiare e precise, buon uso della punteggiatura, specificare che tipo di output vogliamo. Non è da scartare un approccio iterativo per la costruzione del prompt ottimo (approssimazioni successive)

Alcune strategie tipiche sono: sequenze di istruzioni, prompting iterativo o a mini-step, prompt basati sul dialogo, si cerca di forzare il modello a "ragionare" piano o chain of thought (CoT)

Mentre tra i task tipici troviamo:

Summarization: con LLM si ottengono riassunti controllati

Inferenza: ogni richiesta che richiede processing/ragionamento

Trasformazione: traduzione, cambio di stile, cambio di formato

Espansione: da testo breve a testo esteso (dipende anche dalla temperatura, ovvero il livello di creatività)

Ricerca: simil motore di ricerca, ma può generare informazioni non corrette o comunque non aggiornate

Si possono fare richieste senza supervisione (Zero-shot), richieste con esempi di I/O (few-shot) o richieste con step di ragionamento (Chain of thought)